LISTING OF THE CLAIMS

1. (Currently Amended) [[A]]An application specific cache management system stored on a computer readable medium, comprising:

an in-memory database system for managing application specific cached data;

an application utilizing application specific data and having a rule related to caching the application specific data, wherein the application defines the rule[[s]] for the application specific cache;

a wrapper to receive the application specific data from the application and provide at least a portion of the application specific data and a component of the rule to the in-memory database system, wherein the at least the portion of application specific data is the application specific cache data; [[and]]

a rules event table stored in the in-memory database system containing at least one entry comprising a rule type and a reference to the application specific data, the rule type associated with the rule defined by the application; and

an engine ~~operable to monitor the in-memory database system~~ that polls the rules event table, determines that an entry in the rules event table has a rule type that is ready to execute, and applies [[apply]] the rule associated with the rule type of the entry to the application specific cached data referenced by the entry in response to the engine determining that the rule type is ready to execute.[[;]]

~~wherein the engine monitors the in-memory database system and applies the rule to the application-specific cached data without the involvement of the application, or the in-memory database server, or a back-office database.~~

2. (Original) The cache management system of Claim 1, wherein the component of the rule is further defined as a first component and a second component of the rule.

3 and 4. (Cancelled)

5. (Original) The cache management system of Claim 1, wherein the wrapper receives at least a portion of the data from the application and the wrapper provides a part of the at least a portion of the data to the in-memory database system.

6. (Original) The cache management system of Claim 1, further comprising a plug-in operable to communicate with the in-memory database system to receive the data and the component of the rule from the wrapper and communicate the data and information related to the rule to the in-memory database system.

7. (Cancelled)

8. (Original) The cache management system of Claim 1, wherein the rule is defined as an asynchronous rule.

9. (Original) The cache management system of Claim 1, wherein the rule is defined as a synchronous rule.

10. (Previously Presented) The cache management system of Claim 1, wherein the rule includes a refresh data instruction whereby the engine is operable to obtain current data from a database.

11. (Previously Presented) The cache management system of Claim 1, wherein the rule includes a tenure data instruction whereby the engine is operable to release the data after a time period.

12. (Previously Presented) The cache management system of Claim 1, wherein the rule includes a persist data instruction whereby the engine is operable to release the data after a time period unless the data is requested before the expiration of the time period.

13. (Previously Presented) The cache management system of Claim 1, wherein the component of the rule is further defined as a rule type.

14. (Currently Amended) A system stored on a computer readable medium for managing application specific cached data, comprising:

a first application server;

an application operable on the first application server, the application utilizing data and having a rule related to an application specific cache management of the data, wherein the application defines the rules for the application specific cache;

a second application server;

an in-memory database management system operable on the second application server to receive the data;

a wrapper in communication with the application to receive a component of the rule from the application and provide the component of the rule to the in-memory database system; [[and]]

a rules event table stored in the in-memory database system containing at least one entry comprising a rule type and a reference to the application specific data, the rule type associated with the rule defined by the application; and

an engine ~~operable to monitor the in-memory database system~~ that polls the rules event table, determines that an entry in the rules event table has a rule type that is ready to execute, and applies [[apply]] the rule associated with the rule type that is ready to execute to the application specific cached data referenced by the entry.[[;]]

~~wherein the engine monitors the in-memory database system and applies the rule to the application specific cached data without the involvement of the application, or the in-memory database server, or a back-office database. .~~

15. (Previously Presented) The system of Claim 14, wherein the engine is operable on the first application server.

16. (Previously Presented) The system of Claim 14, wherein the wrapper is operable on the first application server.

17. (Previously Presented) The system of Claim 14, wherein the wrapper and the engine are operable on the first application server.

18. (Previously Presented) The system of Claim 14, wherein the engine is operable on the second application server.

19. (Previously Presented) The system of Claim 14, further comprising a third application server and wherein the engine is operable on the third application server.

20. (Currently Amended) A method of managing cached data stored on a computer readable medium, comprising:

obtaining application data and a component of a rule related to the data from an application, wherein the application defines the rule for an application specific cache;

wrapping the application data and the component of the rule;

providing the wrapped application data and component of the rule to an in-memory database server;

storing a rule event in a rules event table stored in the in-memory database server, wherein the rule event comprises a rule type associated with the definition of the rule and a reference to the application data;

~~monitoring~~ polling the rules event table in the in-memory database server;

when the rule event is ready to be executed, applying the rule associated with the rule event to the wrapped application data associated with the rule event based on the rule component; and

caching at least a portion of the wrapped application data according to the rule, to create application cache data;

wherein ~~the application of the rule to the application data occurs~~ the rule event is determined ready to be executed without the involvement of the application, the in-memory database server, or a back office database.

21. (Canceled)

22. (Previously Presented) The method of Claim 20, wherein the rule is defined as an instruction related to a cache management of the data.

23. (Previously Presented) The method of Claim 22, wherein the component of the rule is further defined as a rule type related to the instruction.

24. (Currently Amended) A cache management system stored on a computer readable medium, comprising:

an application utilizing data and having a rule related to caching the data, wherein the application defines the rules for the application specific cache;

an in-memory database management system to receive data;

a wrapper in communication with the application to receive at least a component of the rule; [[and]]

a rules event table stored in the in-memory database system containing at least one entry comprising a rule type and a reference to the data, the rule type associated with the rule defined by the application; and

an engine ~~operable to receive at least the component of the rule from the wrapper and apply the rule to cached data~~ that polls the rules event table, determines that an entry in the rules event table has a rule type that is ready to execute, and applies the rule associated with the rule type of the entry to the application specific cached data referenced by the entry in response to the engine determining that the rule type is ready to execute.[[;]]

~~wherein the engine applies the rule to the cached data without the involvement of the application, or the in-memory database server, or a back-office database.~~

25. (Previously Presented) The cache management system of Claim 24, wherein the data is a refresh data request.

26. (Previously Presented) The cache management system of Claim 24, wherein the rule is an application specific cache data rule.

27. (Currently Amended) The cache management system of Claim 24, wherein the in-memory database management system further includes a storage portion for storing the data utilized by the application ~~and a table operable to maintain a rule event related to the rule for caching data, the rule event pointing to a location in the storage portion of the in memory database where the data related thereto is stored~~.

28. (Previously Presented) The cache management system of Claim 27, wherein the wrapper is further operable to provide at least a portion of the data from the application and a component of the rule to the in-memory database.

29. (Currently Amended) The cache management system of Claim [[28]] 24, wherein the engine is further operable ~~to poll the in-memory database and apply the rule related to the rule event to the data~~ to remove the data referenced by the entry from the in-memory database management system and to remove the entry from the rules event table.

30. (New) The cache management system of Claim 1, wherein the rule defines the functional behavior that is to occur to cache the application specific data.

31. (New) The cache management system of Claim 30, wherein the application defines the rule in source code without guidance from the in-memory database system and writes the rule definition into the application specific cached data using the wrapper.

32. (New) The cache management system of Claim 1, wherein the engine provides manipulations of application specific cache data that are asynchronous with respect to the application.